1.12

INJECTING METEOROLOGY INTO THE GFE SUITE

Thomas J. LeFebvre, M. Mathewson, T. Hansen, and M. Romberg
NOAA Forecast Systems Laboratory, Boulder, CO*

1. INTRODUCTION

The National Weather Service's (NWS) modernization replaced observing systems such as satellite and radar, communication systems, and workstation display systems so that its forecasters could better assimilate these large data sets. With this work nearing completion, NWS would like to capitalize on these new high-resolution data sets by modernizing its public products and services as well. The Interactive Forecast Preparation System (IFPS) promises to do just that by fundamentally changing the way forecasts are defined. In this new paradigm, NWS forecasters would no longer spend the bulk of their forecast shift typing worded forecasts. Instead they would initialize and maintain a set of gridded digital forecasts over the forecast area. From this digital database, routine products in a variety of formats would be generated, from text products to graphical products. Even the raw gridded forecast data could be disseminated to the most sophisticated users.

The GFE Suite is the gridded component to IFPS. Figure 1 illustrates the relationship between the GFESuite, IFPS and the overall Advanced Weather Interactive Processing System (AWIPS). The Graphical Forecast Editor (GFE) allows forecasters to define and manipulate gridded fields of sensible weather (such as temperature, wind, etc.) at the surface. Other applications within IFPS allow the forecaster to extract portions of the data and format it into various products. Numerical model grids, available in AWIPS, are used by the GFE to derive the initial state of the forecast grids before forecasters manipulate them. The balance of this paper focuses on the GFE component of IFPS.

*Corresponding author address: Tom LeFebvre, NOAA Forecast Systems Laboratory, 325 Broadway R/FS6, Boulder CO 80305; e-mail: lefebvre@fsl.noaa.gov

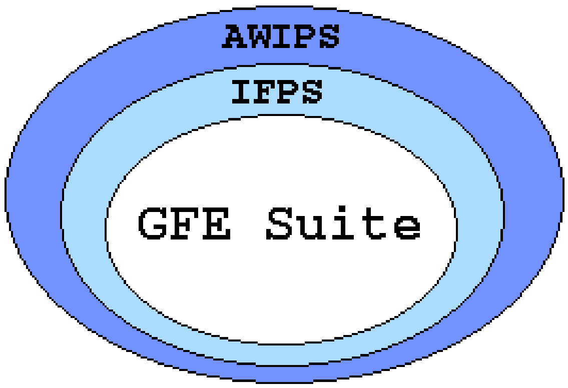

Figure 1. GFESuite's Relationship to IFPS and AWIPS

2. HISTORY

The first few years of our development effort focused on the infrastructure needed to support the GFE such as a client-server database, communications software, and graphical user-interface. During this period the tools used to actually manipulate the gridded fields were simple and tedious to use because they required a significant amount of user input via the pointing device or mouse. The tools had no knowledge of meteorological processes or how the weather elements were intrinsically related to each other by physical laws, which promoted inconsistency between weather elements. Given the initial spatial and temporal resolutions suggested by NWS, forecasters would be required to define and maintain about 3-5 million forecast data values per forecast shift. Clearly, more sophisticated tools were needed to manage such a voluminous data set. These tools had to be flexible enough to simultaneously accommodate climate regimes as diverse as those found in Florida and Alaska. We quickly realized that no single set of tools would suffice for all 120 NWS forecast areas.

The solution was not a set of tools but framework within which forecasters build their own tools to deal with weather situations particular to their local area. This approach works well for many reasons. First, tool can be locally tailored to meet the needs at each office. Second, local weather expertise can be incorporated within the tool allowing new forecasters to quickly take advantage. Third, it would be very difficult for forecasters to communicate all of their requirements to developers. Dozens of iterations would be required before the system was efficient enough for forecasters. Finally, expressing the forecast digitally is a large paradigm shift for forecasters. For any system to be accepted in this exploratory environment, a large degree of user-configurability is essential. Since the tools built within this framework could be programmed with meteorological intelligence, we name it Smart Tools.

3. SMART TOOLS

The Smart Tool framework evolved from our work on the user-interface. Several years ago, we completely renovated our user-interface component using a new scripting language called Python. It quickly became apparent that we could develop code nearly ten times faster in Python than in our other language C++. Python's simplicity coupled with the knowledge that most Smart Tools would generally take the form of simple arithmetic expressions led us to believe that we could build a Python-based framework that forecasters could learn quickly. In addition, since the Python is a scripting language and no compilation is required, they are easily shared among forecast offices.

Figure 2 conceptually illustrates how Smart Tools relate to the other objects and data within the GFE and AWIPS. Any Smart Tool has access to the full suite of AWIPS model grids, the entire set of GFE forecast grids, as well as a 1 km resolution topography data set. These data sets may be utilized in any combination, allowing the forecaster to build tools with very few limitations.

Smart Tools have full access to all of the synoptic-scale models available from the AWIPS data server in 3 dimensions plus time. These include Eta, AVN, NGM, MRF, as well as any local models running at a particular office. Using these gridded data sets forecasters can write tools that derive surface sensible weather elements directly or adjust already defined weather elements. Models may be used in any combination to provide a model-blending capability that would use two or more models to arrive at a consensus. High-resolution local models have the potential of providing highly detailed forecasts.

Providing access to the GFE forecast grids enhances the forecaster's ability to create consistency among forecast weather elements that are intrinsically related. For example, once the "weather and obstructions to vision" element has been defined, a Smart Tool could then define

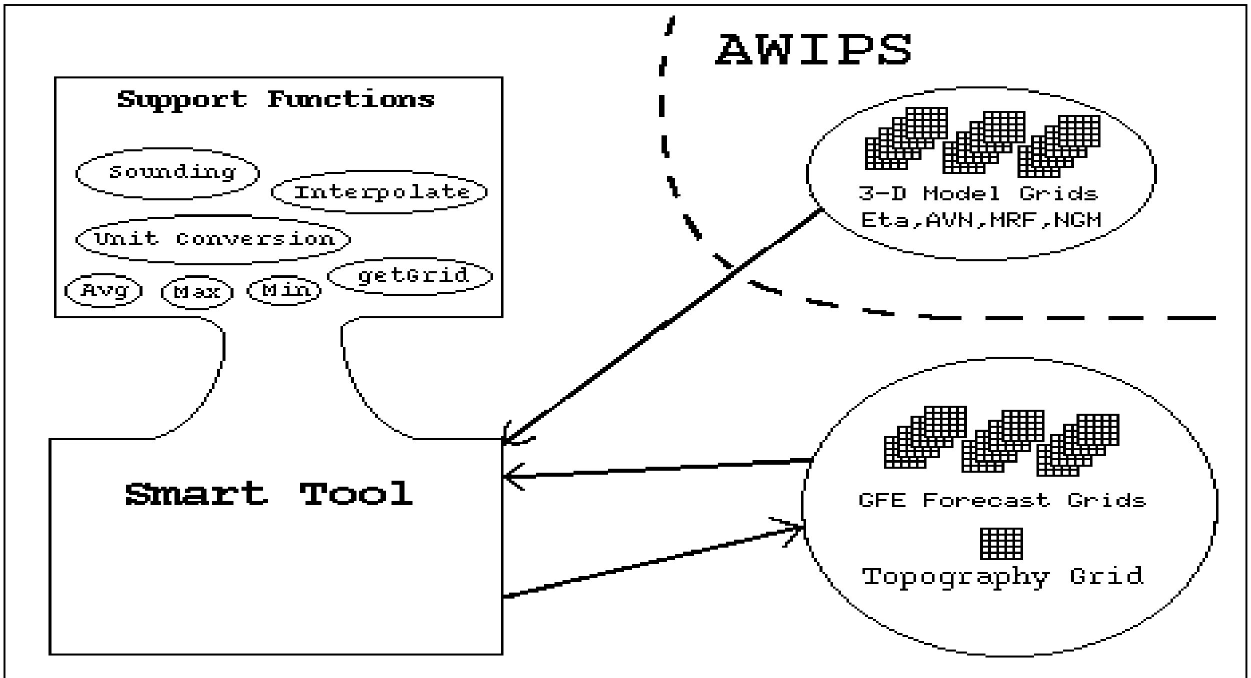

Figure 2. Relationship of Smart Tools, Support Functions, AWIPS model grids, and GFE Forecast Grids.

the clouds, quantitative precipitation, and probability of precipitation based on the weather type and intensity while maintaining good consistency between these elements. Once the forecast is complete, Smart Tools running as consistency checkers can scan the forecast grids and determine areas that violate particular rules such as precipitation with no clouds or temperature lower than dew point at each forecast point. These consistency checkers can either identify areas that need forecaster attention or actually modify the data to make the forecast consistent.

A built-in 1 km topography data set gives Smart Tools the potential to define local terrain-driven phenomena in areas where terrain plays a significant role in the weather. For example, a gridded surface temperature forecast derived from a synoptic-scale model with coarse terrain can be adjusted based on the elevation at each grid point.

The result is a highly detailed temperature grid that accurately reflects the influence that elevation has on the temperature (Wier 1996). In areas where the precipitation pattern is determined by terrain, Smart Tools can calculate those areas where the combination of wind and elevation enhances vertical motion and add detail to the quantitative precipitation forecasts. Access to high-resolution topography data allows Smart Tools to add local effects in forecast areas where terrain plays a role in the weather.

The support functions indicated in Figure 2 represent high-level utility functions that help forecasters build Smart Tools more easily. For example, the function makeSounding reads data from the specified model and returns a vertical profile of the specified model variable. Another function returns an interpolated value at a specified level. Other functions convert values from to a different set of units such as m/s to MPH. These functions simply reduce the amount of code required for a particular Smart Tool and hence the burden on forecasters to know a great deal about how to access the data. We expect the number of these kinds of functions to grow significantly with time.

3.1. SMART TOOL EXAMPLES

Figure 3 is an example of a Smart Tool that calculates the surface temperature based on the temperature sounding extracted from model grids. First the sounding levels are defined and the sounding retrieved by the makeSounding function. Next the elevation is converted from feet to meters. Then a value is extracted from the sounding at the elevation, converted to degrees Fahrenheit, and returned.

```
# Make a sounding for T at this point
# Height will increase in the sounding

levels = ["MB1000","MB950","MB900","MB850",
   "MB800","MB750","MB700","MB650","MB600"]

sounding = self.makeSounding(levels, x, y,
        GridTimeRange, "t", "Eta")

topo_M = Topo/3.28084

actualT_K = self.getValueFromSounding(sounding,
        topo_M, self.interpolateValues)

actualT_F = self.convertKtoF(actualT_K)

return actualT_F
```

Figure 3. Smart Tool to Extract Surface Temperature from Model Data

Figure 4 illustrates a Smart Tool that calculates Snow Accumulation based on the topography, quantitative precipitation forecast (QFP), and the freezing level. The first step is to define a

```
SnowRatioDictionary = {9:18,10:18,11:17,
12:17, 13:16, 14:15,15:14, 16:13, 17:13,
18:12,19:12,20:11,21:11,22:11,23:10,
24:10,25:9,26:9,27:8,28:8,29:8}

def SnowAmt_SmartTool(QPF, T, FzLevel, Topo):

"Determines SnowAmt from QPF, T,
 FzLevel and Topo information"

   if T < 9:
      SnowRatio = 20
   elif T >= 30:
      SnowRatio = 7
   else:
      SnowRatio = SnowRatioDict[int(T)]

   # Determine new value
   if (FzLevel-1000) <= Topo:
      SnowAmt = SnowRatio * QPF
   else:
      SnowAmt = 0
```

Figure 4. Smart Tool to Calculate Snow Amount from QPF, Freezing Level, Temperature, and Topography

relationship between the surface temperature (already defined) and the snow ratio. Next the surface temperature determines this ratio. If the surface elevation is 100 feet above the freezing level, the snow amount is calculated based on the QFP and the snow ratio and returned.

4. FUTURE WORK

4.1. SMART IFPINIT

Currently the GFE includes a facility called IFPInit to derive surface weather elements over the forecast area as an initial forecast (Wier 1995). But since IFPInit is written in C++, forecasters have no control over the algorithms that derive these surface forecasts. In the future, we plan to convert this facility to Python so that field forecasters can adjust the algorithms for their local area much like the way Smart Tools work today. As better algorithms are written, they can be easily share amongst forecast offices, since they are implemented in a scripting language.

4.2. FORECAST MONITORING

One critical data set that is currently missing from the GFE is surface observations. With access to observations, Smart Tools could be developed that monitor the observations, compare them to the current gridded forecast and inform the forecaster when there is a significant discrepancy. In the coming months, we plan to incorporate observations so that the GFE system can perform these forecast monitoring functions.

4.3. FORECAST VERIFICATION

Once surface observations are incorporated into the GFE, verification functions could be written to provide feedback with respect to how well the observations correlate with the forecast. These verification routines would not only apply to the official forecast, but also forecasts generated from the Smart IFPInit algorithms. This would give forecasters the ability to assess the accuracy of a particular model or algorithm before incorporating it as part of the official forecast. Using this technique, new algorithms developed at a local office could be evaluated for accuracy against archived data before they are incorporated as part of the official forecast.

5. CONCLUSION

The tools offered by early versions of the GFE were very simple and unaware of meteorological concepts. The introduction of Smart Tools has changed this dramatically. Smart Tools give forecasters the capability to think in of some concept meteorological terms, express that concept to the GFE system, and modify the gridded forecast accordingly. For example, if a forecaster thinks, "Fog in low-lying valleys", a Smart Tool can be written to express that concept and modify the data in very high detail.

The introduction of Smart Tools represents a fundamental change in the methodology forecasters use to express the forecast. Rather than spend hours per shift typing, forecasters can express high-level meteorological concepts to the GFE and see them realized in the data. While forecast information disseminated from the weather is largely qualitative now, forecasting with a gridded methodology changes this to mostly quantitative. These gridded data sets will also change the way consumers use weather forecast information. With the forecasts expressed in gridded digital form, NWS can then offer a wide variety of modernized products and services (LeFebvre, 1996).

6. REFERENCES

Hansen, T. L., Mark A. Mathewson, Mike Romberg, 2000: Forecast Methodology using the GFESuite Preprints, *Seventeenth Conference on Weather Analysis and Forecasting*, Norfolk VA, Amer. Meteor. Soc.

LeFebvre, T. J., Corby A. Bacco, and Mike Romberg, 1996: Gridded Forecast Products and Services Under Development at FSL. Preprints, *Fifteenth Conference on Weather Analysis and Forecasting*, Norfolk VA, Amer. Meteor. Soc.

LeFebvre, T. J., 1995: Operational Forecasting With AFPS. Preprints, *Eleventh International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology*, Dallas TX, Amer. Meteor. Soc.

Mathewson, M. A., 1996: Using the AWIPS Forecast Preparation System (AFPS). Preprints, *Twelfth International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology*, Atlanta GA, Amer. Meteor. Soc.

Wier, S. K., and J. S. Wakefield, 1996: Using numerical model output to provide initial forecasts of surface weather for the AFPS. Preprints, *Twelfth International Conference on Interactive Information and Processing*

Systems for Meteorology, Oceanography, and Hydrology, Atlanta GA, Amer. Meteor. Soc.

Wier, S. K., 1995: Interpolating Between Grids of Meteorological Data for AFPS. Preprints, *Eleventh International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology*, Dallas TX, Amer. Meteor. Soc.